FMC32　スタートアップガイド

(by PUSB3503)

Nippon Pulse Motor Co., Ltd.

— **Blank Page** —

目　次

1. はじめに 1
２．概要 2
2-1. 今回作成する動作 2
2-2. 必要な環境 4
2-3. 接続例 4
2-4. 電源投入の順番 5
2-5. 動作を作成する手順の概要 5
３．動作の作成 6
3-1. プログラムの起動 6
3-2. プロジェクトの作成 7
3-3. 動作パターンの設計 8
3-3-1. 動作パターン作成画面の表示 8
3-3-2. 原点復帰動作の作成と登録 9
3-3-3. CW 方向へ **50** パルス出力する動作の作成と登録 11
3-3-4. CW 方向へ **1000** パルス出力する動作の作成と登録 13
3-3-5. CCW 方向へ **2000** パルス出力する動作の作成と登録 16
3-3-6. 登録した動作パターンデータの保存 18
3-3-7. 登録したデータの確認 19
3-4. 動作手順のプログラミング 20
3-4-1. エディタ画面の表示 20
3-4-2. プログラミング作業の流れ 21
3-4-3. プログラムの動作確認（シミュレーション） 31
3-4-4. プログラムの保存 34
3-4-5. プログラムの動作確認（実動作） 35
3-4-6. EEPROM へのデータの書込み 37
3-4-7. スタンドアロンでの実行 38

1. はじめに

本説明書では、FMC32 を利用したモータ制御のための基本的な操作の流れや作業に関する解説を行います。

ここでは、ハードウェアおよびソフトウェアのインストール作業が完了しているものとして説明をおこないます。

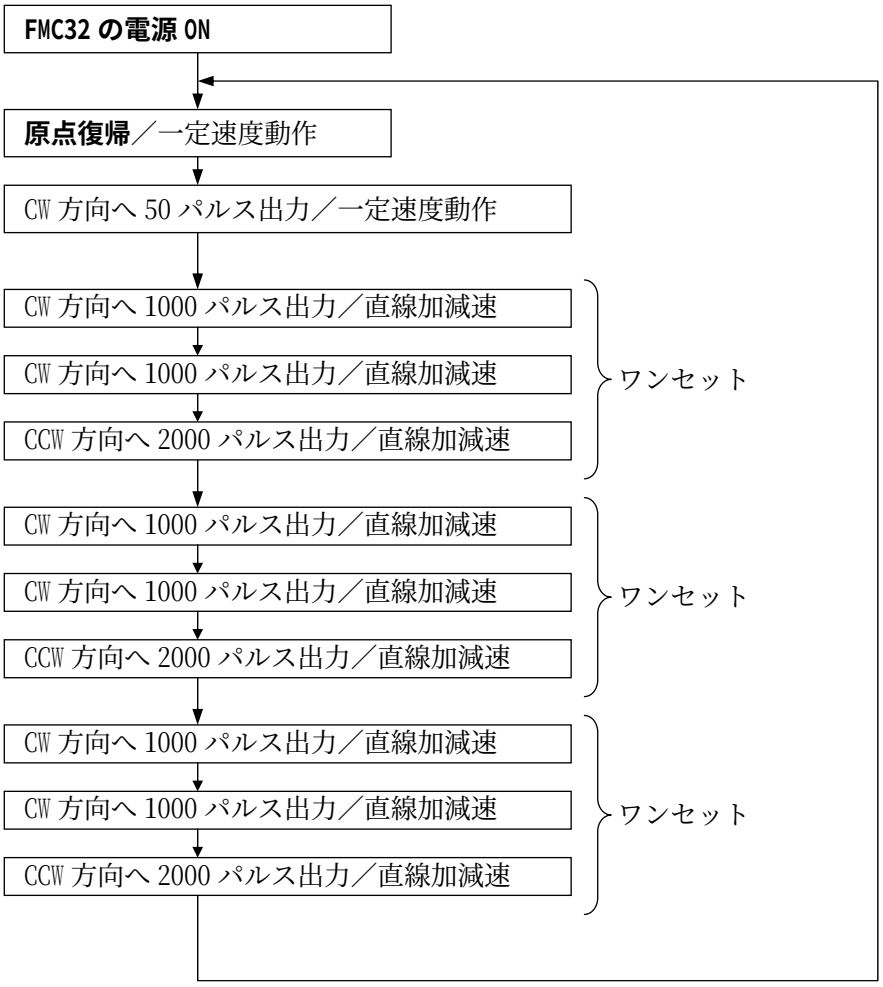
尚、FMC32 の取り扱いに関する詳細に付いては、別途弊社の取扱説明書(下記に記載)を合わせて御参照ください。

▼ドライバ内蔵小型コントローラ FMC32 ハードウェア取扱説明書・・・・・・・・・・・・資料№YA7162
▼ドライバ内蔵小型コントローラ FMC32 制御ソフトウェア取扱説明書・・・・・・・・資料№YA7163
▼USB↔4線式シリアル変換ユニット PUSB3503 取扱説明書・・・・・・・・・・・・・・・・・・・・・・資料№YA7164
▼シリアルバス制御パルスコントロール LSI PCD2112 ユーザーズマニュアル・・・資料№DA70115

2．概要

2-1．今回作成する動作

本書では、FMC32 を使用して次に示すような簡単な動作を制御するための手順を解説します。
FMC32 に行わせる次のような一連の動作を、"実行順序プログラム"(もしくは動作シーケンス) と呼びます。
また実行順序プログラムを作成する作業をプログラム作成（プログラミング）と呼び、実行順序プログラムを記述したデータをプログラムと呼びます。



動作の概要：

電源投入後、原点を探し、原点から 50 パルス分のスタート位置まで移動する。
続いて正転～逆転まででワンセットの動作を3回繰り返した後､再度原点を探す動作を繰り返す。

その他の条件：

初速度は 10pps、最高速度は 500pps とする。
加速時間と減速時間は一律 500ms とする。
一定速度は 200pps とする。
ワンセット内の各動作間に、200ms の待ち時間を挿入する。

2-2. 必要な環境

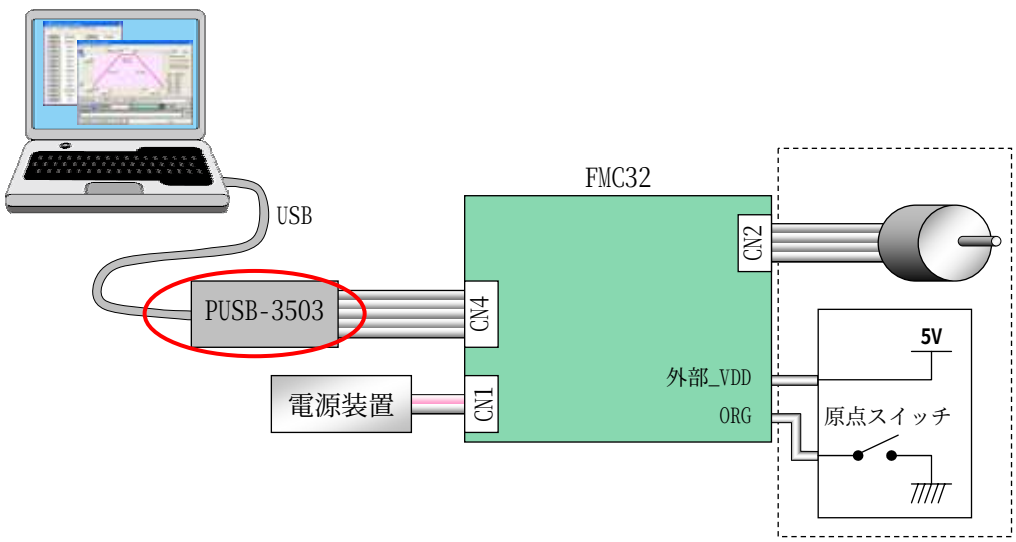
本書で作成する動作を実行するための環境が必要となります。

項目	内容	備考
PC	動作パターンとプログラム作成、およびプログラムのシミュレーション、動作データの FMC32 への書込みで使用します。 OS は WindpwsXP SP3、もしくは Windows7 を対象としています。	必須
PUSB-3503	PC から FMC32 を制御するときに使用する弊社製品です。	必須
FMC32	本書で操作する製品です。	必須
FMC32 用電源装置	FMC32 へ供給する電源で、11V～26V まで対応しています。 接続するモータの駆動電源でもあるので、モータにあった電源装置をご用意ください。	必須
ステッピングモータ	FMC32 でドライブできるモータ。 弊社製モータ「ＰＦＣＵ２５－２４Ｃ１Ｇ」等。	
原点スイッチ	FMC32 に対して原点を知らせるもので、トグルスイッチなどでも代用できます。 原点センサ付のスライダなどがあれば理想的です。	
原点スイッチ用電源	原点スイッチと、FMC32 のフォトカプラを駆動させるための 5V 電源です。	

備考欄が空白になっている部品または装置は、用意できなければ無くてもかまいません。用意できない場合でも、プログラムした動作のシミュレーションによる確認までは行えます。

2-3. 接続例

次のように接続します。



点線で囲まれた部分は、用意できた場合のみでかまいません。

また原点スイッチは非導通状態にしてください。FMC32 は、デフォルトではスイッチが入った状態を原点と判断します。

2-4. 電源投入の順番

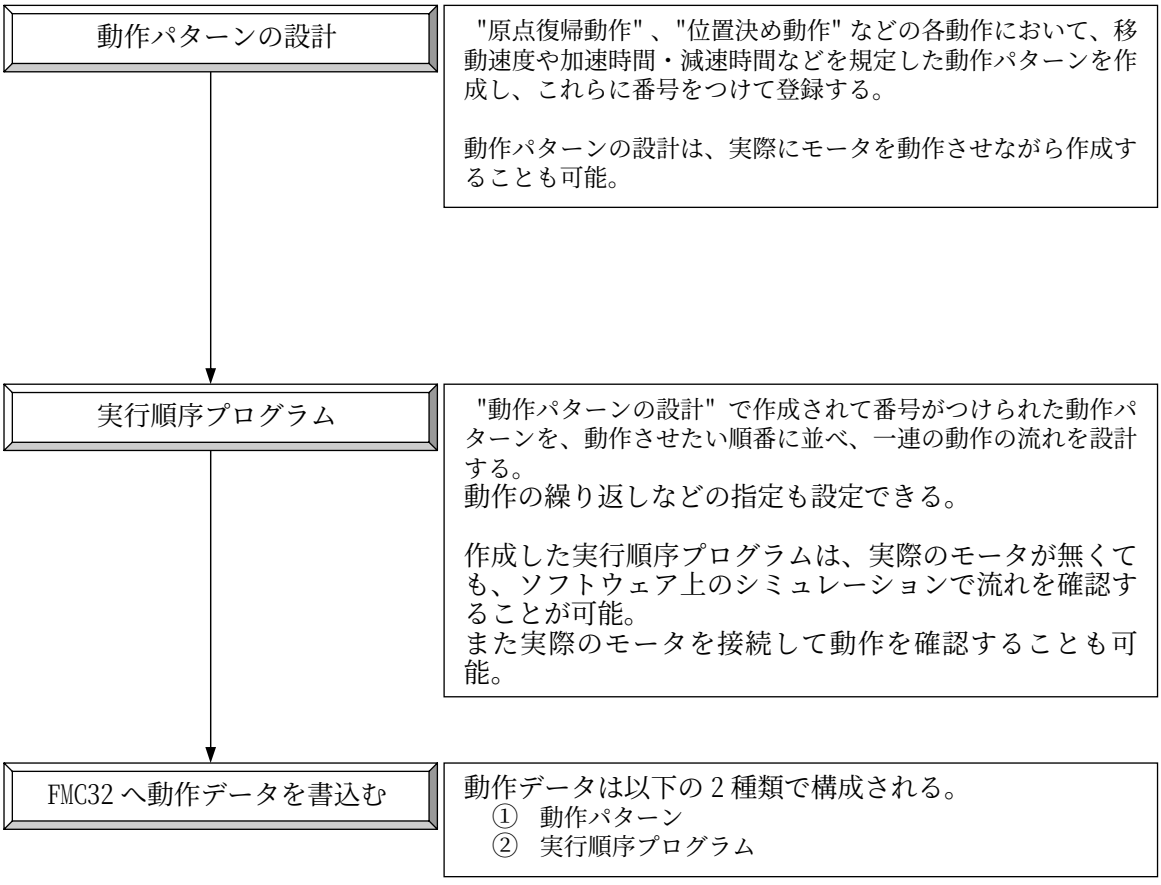
電源を投入する順番は次のような順番でお願いします。

① FMC32 と PUSB-3503 を接続、および FMC32 と原点スイッチ、モータを接続。
② FMC32 と原点スイッチ(5V)の電源を投入。この 2 つに順番はありません。
③ PUSB-3503 と PC を USB ケーブルで接続（PUSB-3503 は USB のバスパワーから電源をもらいます。)。

電源の切断は、投入順番と逆でお願いします。

2-5. 動作を作成する手順の概要

FMC32 への実行順序プログラムの設計は、次のように 3 つに大別できます。

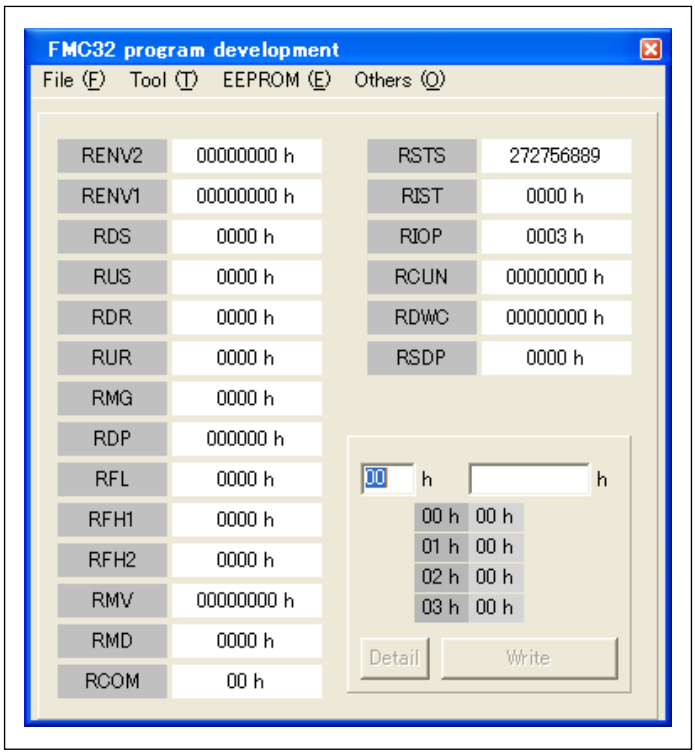


3．動作の作成

3-1．プログラムの起動

次のアイコンをダブルクリックして下さい。

ソフトウェアが起動し、以下のメイン画面が表示されます。

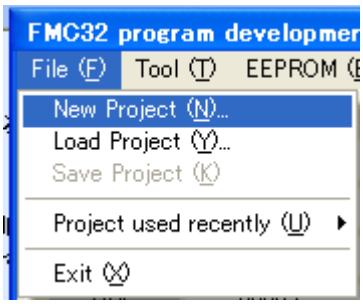


【メイン画面】

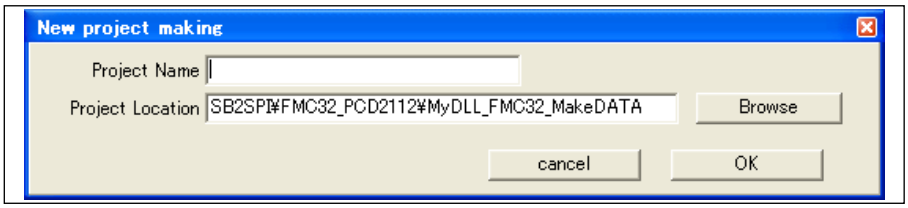
3-2. プロジェクトの作成

まず作成したデータを格納するプロジェクトを作成します。

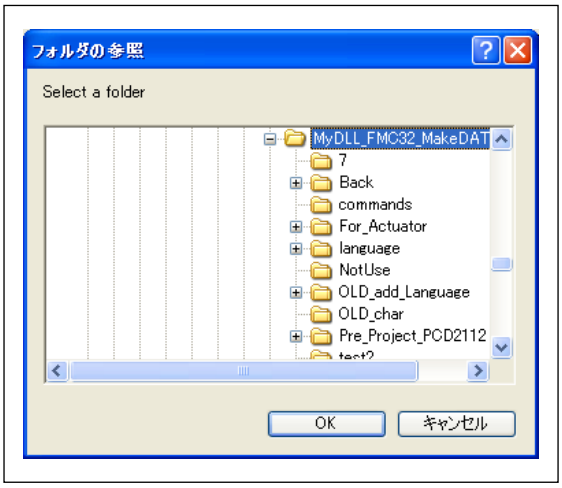
①メニューの "ファイル" (File) → "新規プロジェクト" (New Project...)をクリックします。



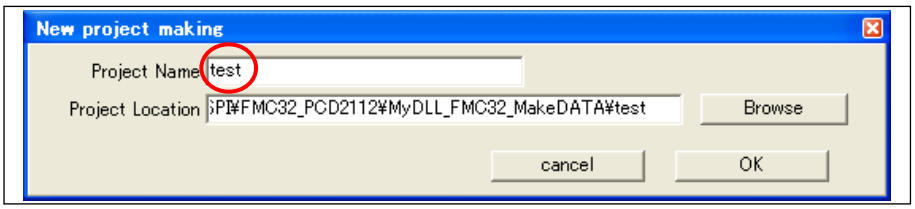
次の画面が表示されます。



② "Browse" をクリックし、プロジェクトを格納するフォルダを選択してください。
ここで選んだフォルダの下に、別途指定するプロジェクト名と同じ名前のフォルダが作成され、その中に動作データ等が格納されます。



③プロジェクト名を入力します。
入力後、 "OK" をクリックしてください。

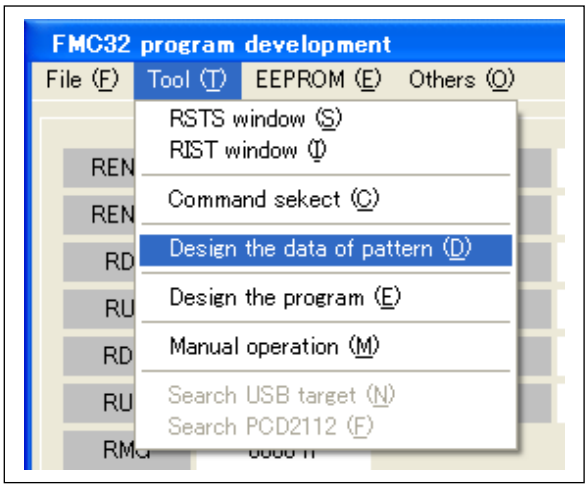


3-3. 動作パターンの設計

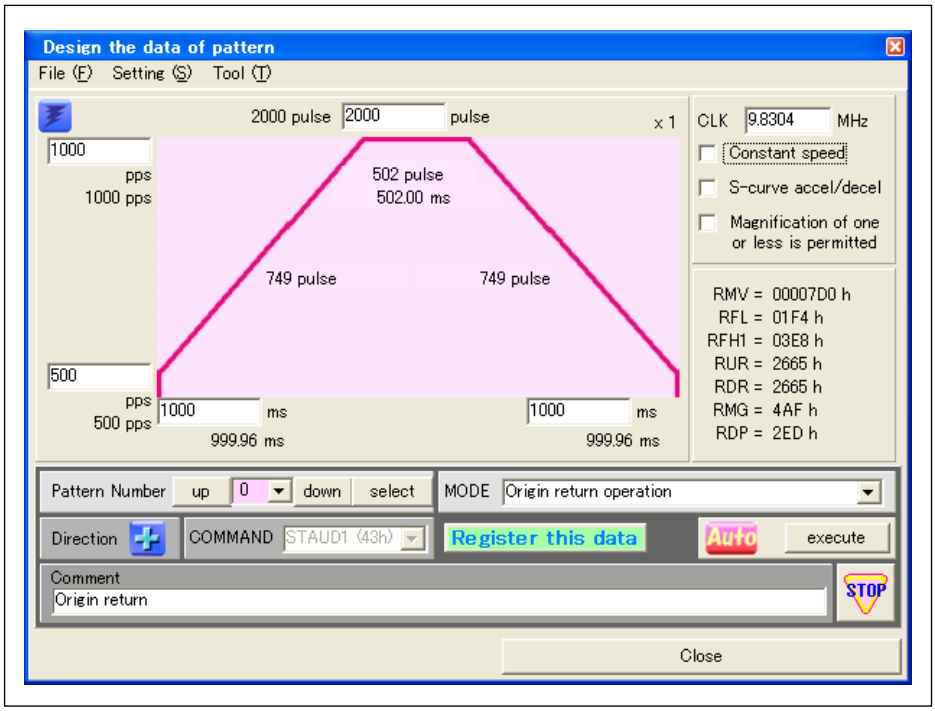
ここで言う動作パターンとは、いわゆる位置決め動作や原点復帰動作などを指します。この位置決め動作に必要な「初速度」「最高速度」「加速時間」「減速時間」「移動量」などは、実際にモータを動作させながら設定することができます。

3-3-1. 動作パターン作成画面の表示

「Tool」メニューの、「動作パターンの設計」(Design the data of pattern)をクリックしてください。

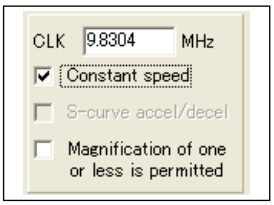


以下の動作パターンの設計画面が表示されます。

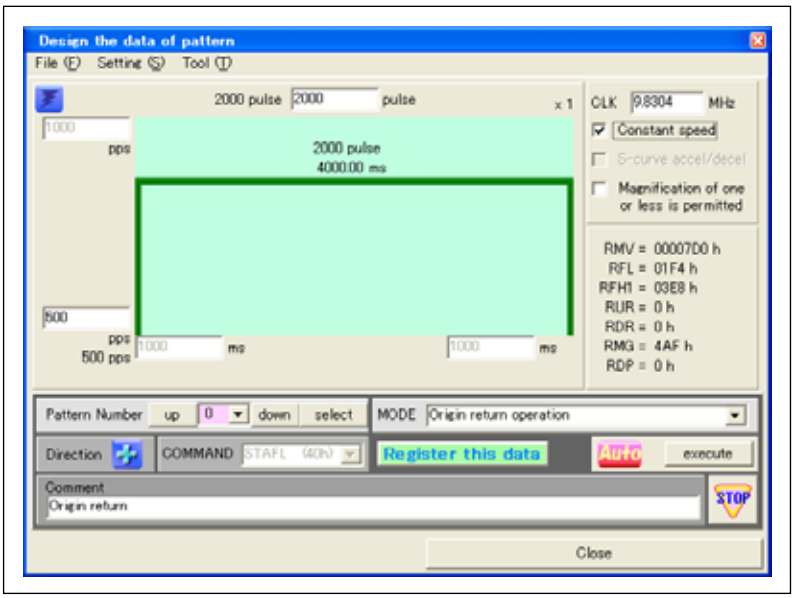


3-3-2. 原点復帰動作の作成と登録

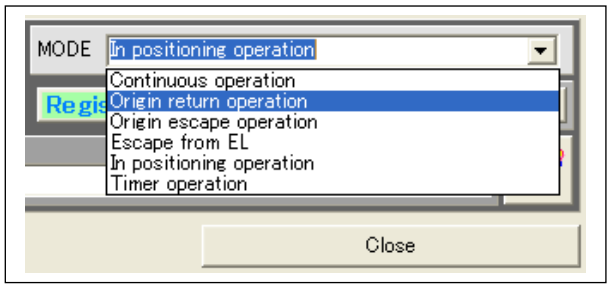
設計画面の "一定速度" (Constant speed)にチェックを入れます。



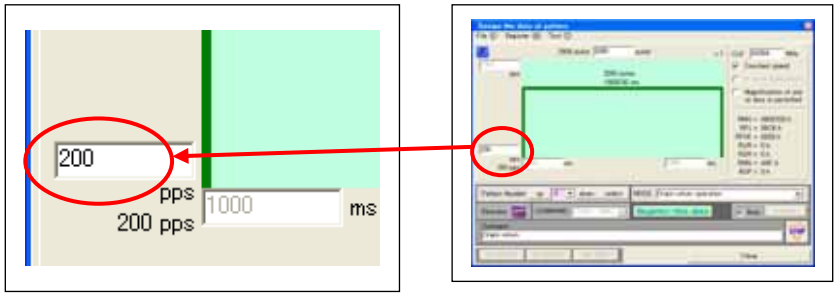
設計画面が以下のようになります。



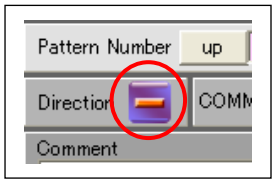
さらに "MODE" の横のプルダウンメニューから "原点復帰動作" (Origin return operation)を選択してください。



つづいて以下の場所の数値を "200" に変更してください。

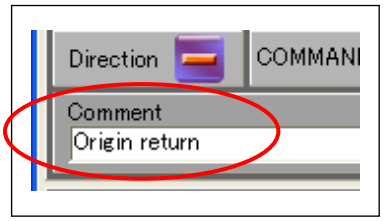


次に "Direction" の横のアイコンをクリックし、マイナスに変更します。



以上で原点復帰動作の動作パターンが作成できました。

また、できるだけコメントを入力してください。後のプログラム作業が楽になります。



[動作の確認]

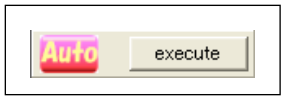
実際にモータが接続されている場合、そのモータを動作させることができます。これにより動作パターンの確認ができます。

以下のボタンをクリックして、FMC32 を励磁オン状態に切り替えてください。

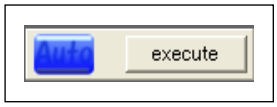
励磁状態はオフ

励磁状態はオン

以下の「execute」ボタンをクリックしてください。
モータが動作を開始した後、原点スイッチがＯＮするとモータが停止します。



Auto 有効

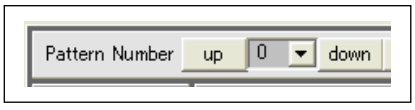


Auto 無効

注 1：Auto は有効のままにして下さい。
注 2：原点スイッチが無い場合は、原点復帰動作の確認はできません。
注 3：モータが接続されていない場合、メイン画面の "RCUN" レジスタの値の増減で、動作状況を確認できます。

[パターン番号の確認]

パターン番号がゼロであることを確認してください。

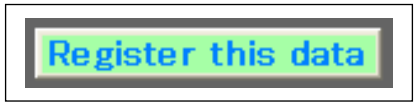


[動作パターンの登録]

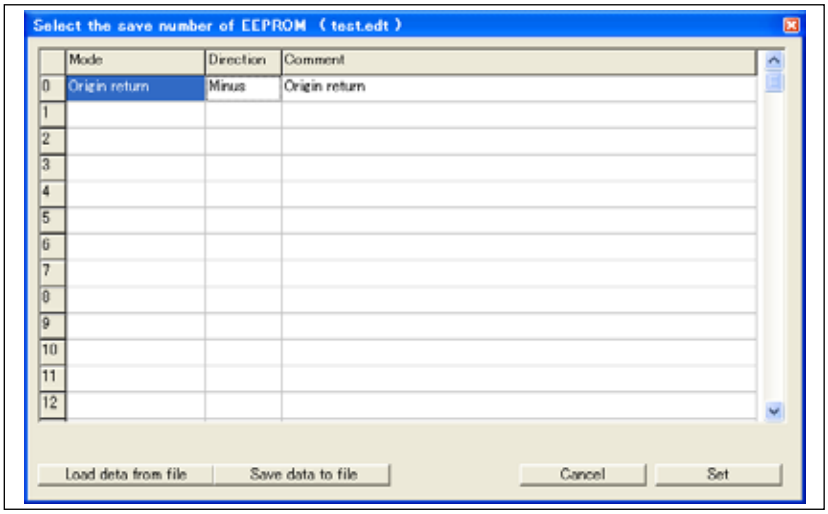
作成した動作パターンを登録します。登録作業を行わなければ、後に行うプログラムから利用できません。基本的には、動作確認して問題が無かった場合に、そのデータを登録して下さい。

動作パターンは、最大 32 個まで登録できます。

以下の「Register this data」ボタンをクリックしてください。



次のような画面が表示されます。

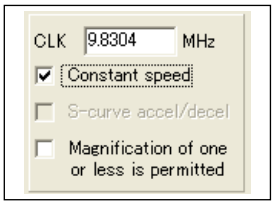


作成したデータの "動作モード" 、"動作方向" 、"コメント" が表示されます。それ以外の詳細な情報は表示されないので、コメントで判断できるようにしてください。

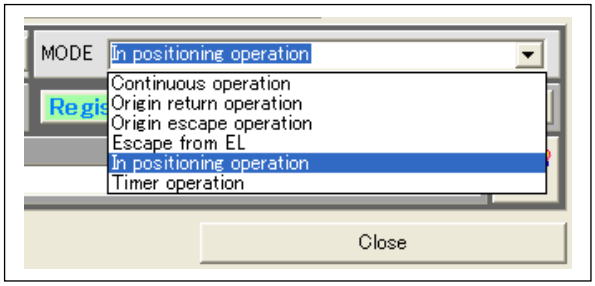
ここで "Set" をクリックしてください。これでパターン番号ゼロに原点復帰動作が登録されます。

3-3-3. CW 方向へ 50 パルス出力する動作の作成と登録

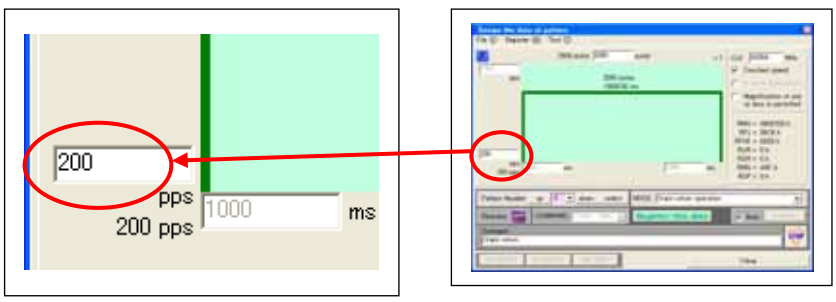
設計画面の "一定速度" (Constant speed)のチェックは入れたままにしてください。



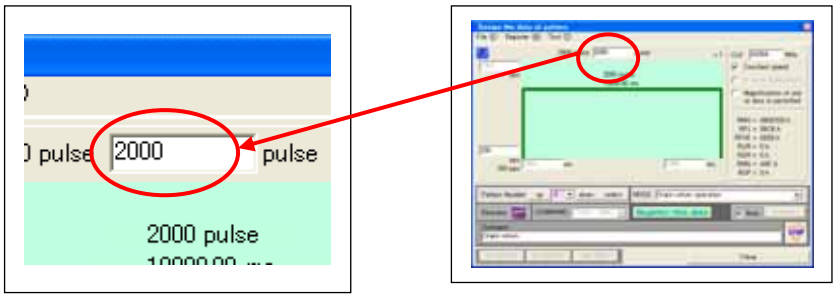
次に "MODE" の横のプルダウンメニューから "位置決め動作" (Inpositioning operation)を選択します。



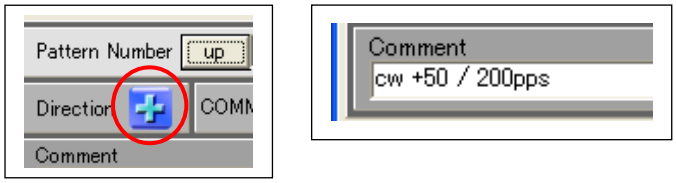
以下の場所の数値は "200" のまま変更はありません。



以下の値を "50" に変更してください。これが出力されるパルス数となります。



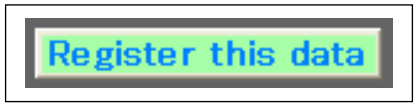
次に "Direction" の横のアイコンをクリックし、プラスに変更します。
コメントも入力しましょう。



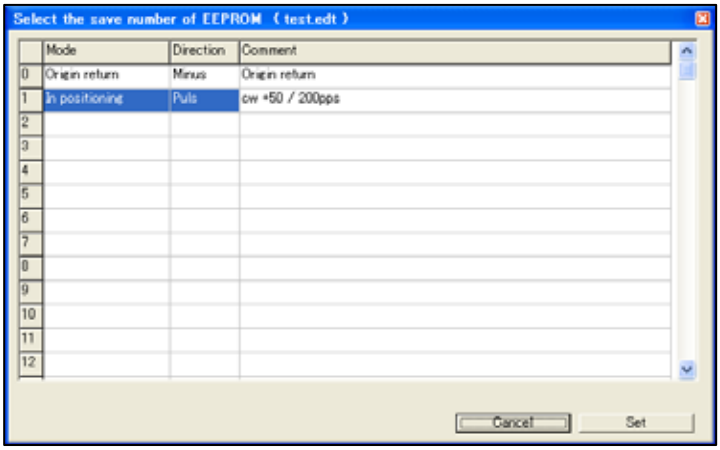
以上で CW 方向へ 50 パルス出力する動作が作成できました。

続いて作成したパターンを登録するのですが、登録前に "Pattern Number" の横の "up" をクリックし、登録番号を "1" に変更してください。以下の図のように、登録番号の背景色がグレーの場合、動作が未登録であることを表します。

以下の「Register this data」ボタンをクリックしてください。

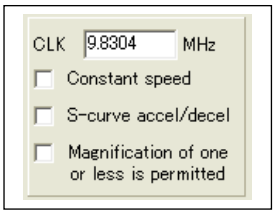


ここで "Set" をクリックしてください。これでパターン番号 1 に動作が登録されます。

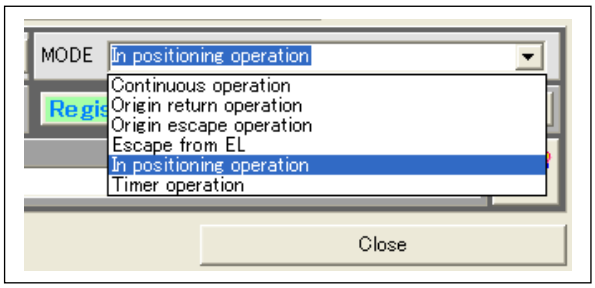


3-3-4. CW方向へ1000パルス出力する動作の作成と登録

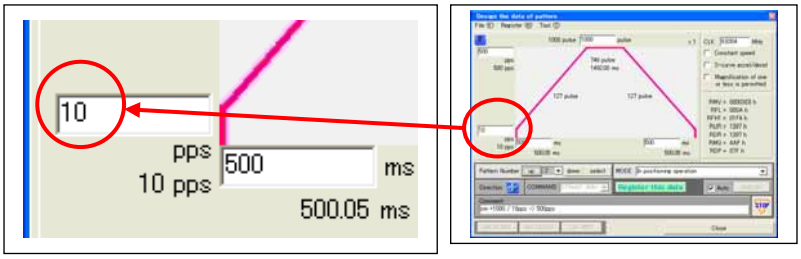
設計画面の "一定速度" (Constant speed)のチェックをはずしてください。



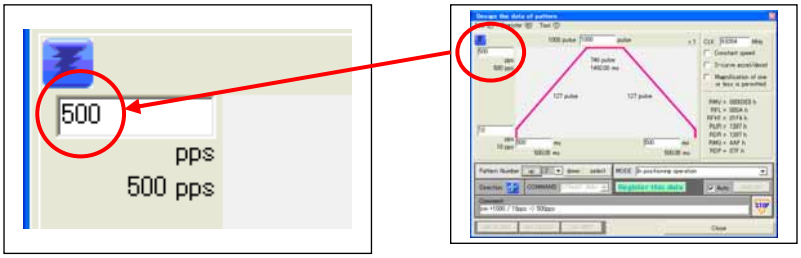
次に "MODE" の横のプルダウンメニューから "位置決め動作" (Inpositioning operation)を選択します。



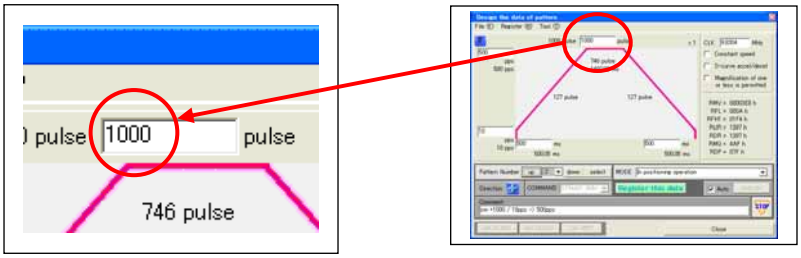
以下のように初速度を設定します。



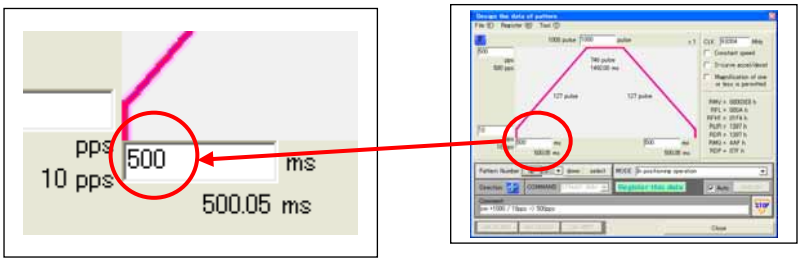
最高速度に次の値を設定します。



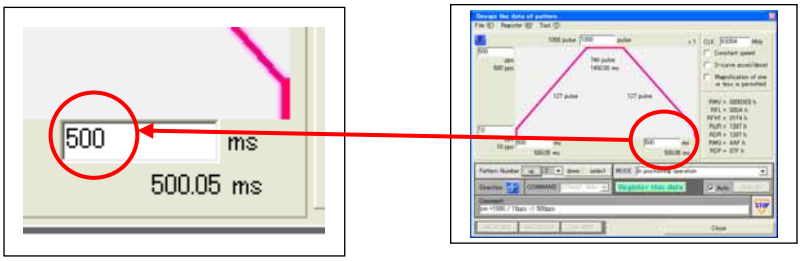
出力するパルス数を設定します。



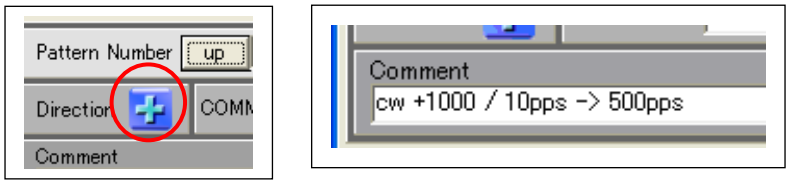
加速時間を設定します。



減速時間を設定します。



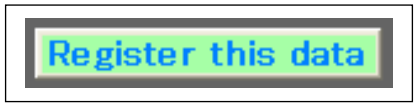
次に "Direction" の横のアイコンをクリックし、プラスに変更します。
コメントも入力しましょう。



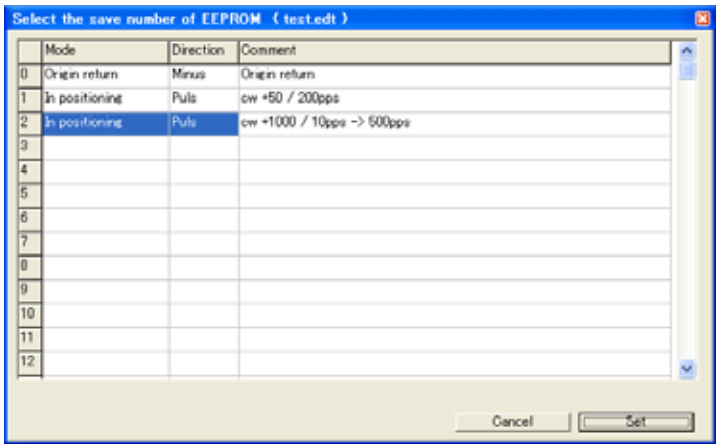
以上で CW 方向へ 1000 パルス出力する動作が作成できました。

続いて作成したパターンを登録するのですが、登録前に "Pattern Number" の横の "up" をクリックし、登録番号を "2" に変更してください。

以下の「Register this data」ボタンをクリックしてください。



ここで "Set" をクリックしてください。これでパターン番号 2 に動作が登録されます。

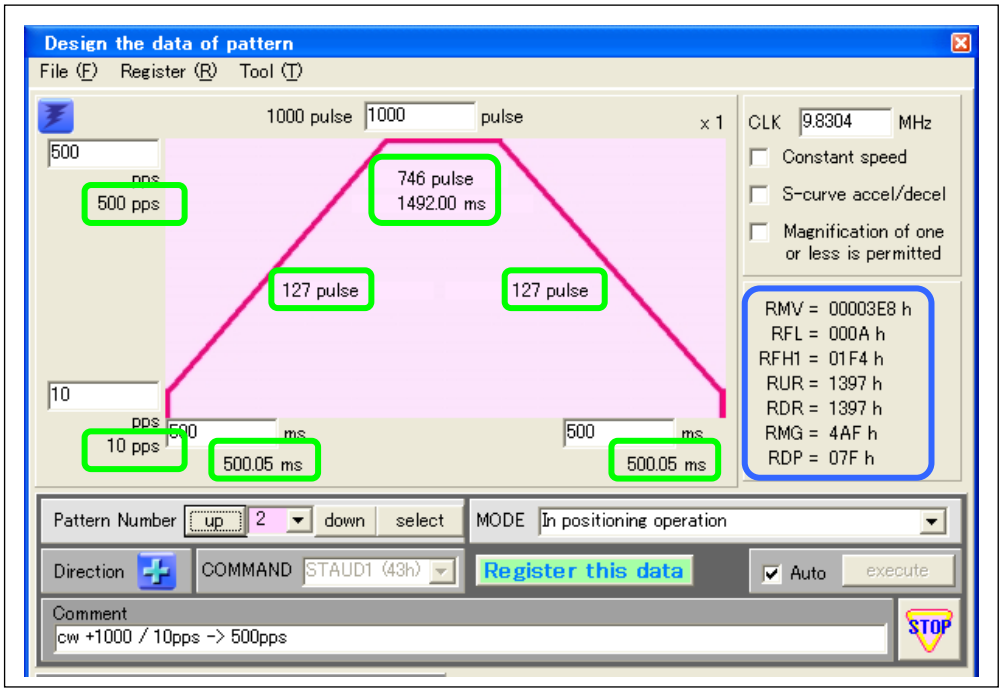


[設定した値の確認]

ここまでの操作で、動作パターンの設計画面は以下のようになっているはずです。

FMC32 には PCD2112 が使用されていますが、この PCD2112 が指定された値で動作するように各種レジスタへの値の設定が行われます。

以下の図で、青線で囲まれた部分に PCD2112 のレジスタへ設定される実際の値が表示されています。

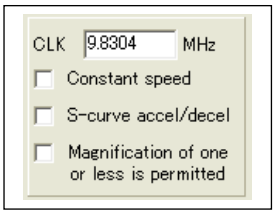


緑線で囲まれた値は、レジスタへ設定された値によって、実際に PCD2112 がどのように動作するかを示したものです。

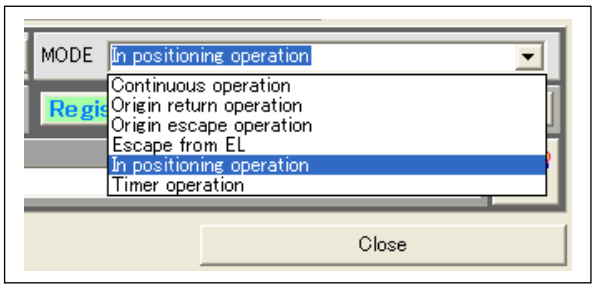
必ずしも設定した値通りの動作を行うとは限りません。これは PCD2112 の仕様上、PCD2112 の内部計算結果と設定値との間で、誤差が発生するからです。例えば、加速時間として 500ms を指定したのですが、計算上では 500.05ms かかります。

3-3-5. CCW 方向へ 2000 パルス出力する動作の作成と登録

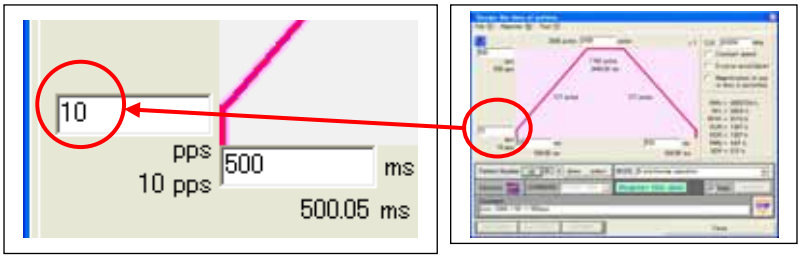
設計画面の "一定速度" (Constant speed)のチェックははずしたままにしてください。



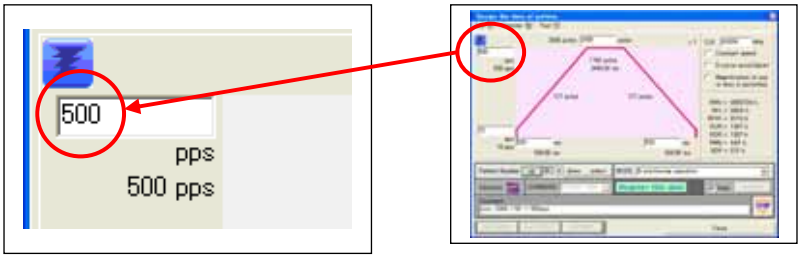
次に "MODE" の横のプルダウンメニューから "位置決め動作" (Inpositioning operation)を選択します。



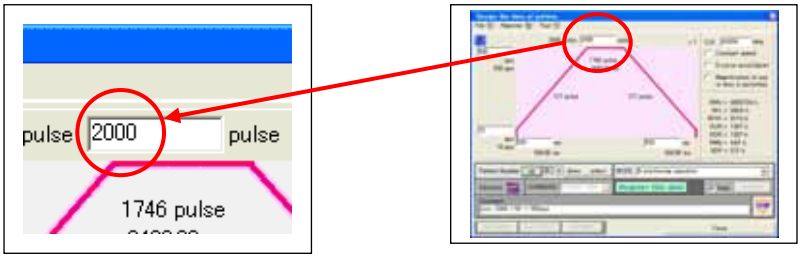
初速度は変更ありません。



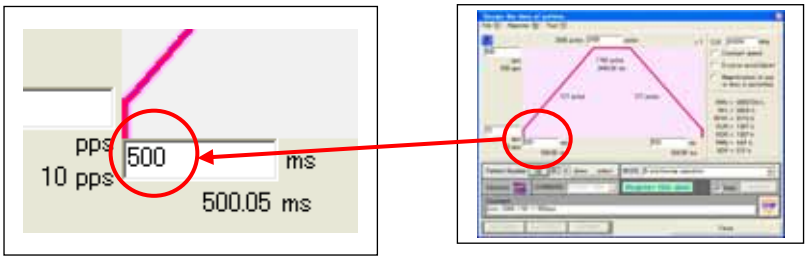
最高速度も変更ありません。



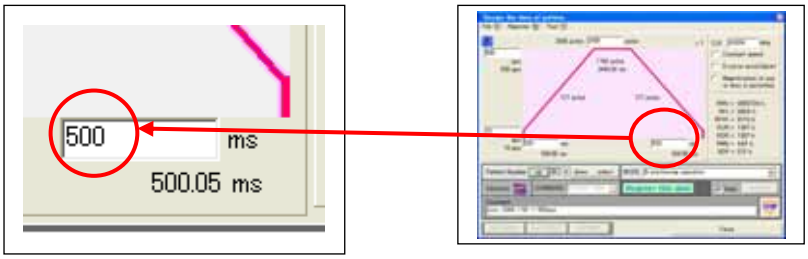
出力するパルス数を設定します。



加速時間は変更ありません。



減速時間は変更ありません。

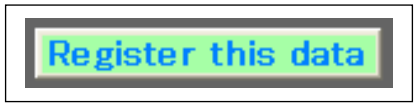


次に "Direction" の横のアイコンをクリックし、マイナスに変更します。
コメントも変更してください。

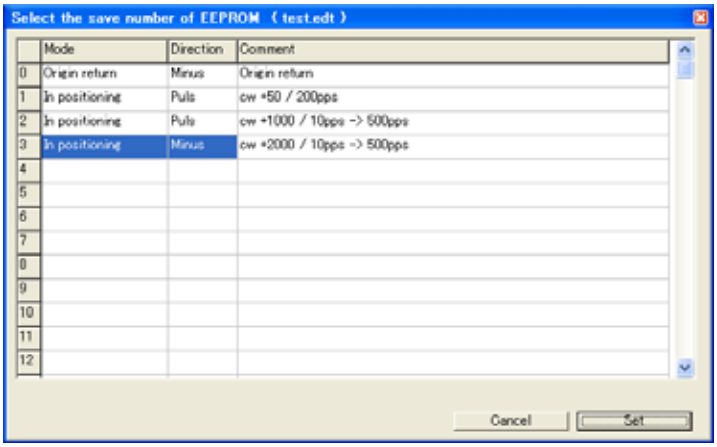
以上で CCW 方向へ 2000 パルス出力する動作が作成できました。

続いて作成したパターンを登録するのですが、登録前に "Pattern Number" の横の "up" をクリックし、登録番号を "3" に変更してください。

以下の「Register this data」ボタンをクリックしてください。



ここで "Set" をクリックしてください。これでパターン番号 3 に動作が登録されます。



3-3-6. 登録した動作パターンデータの保存

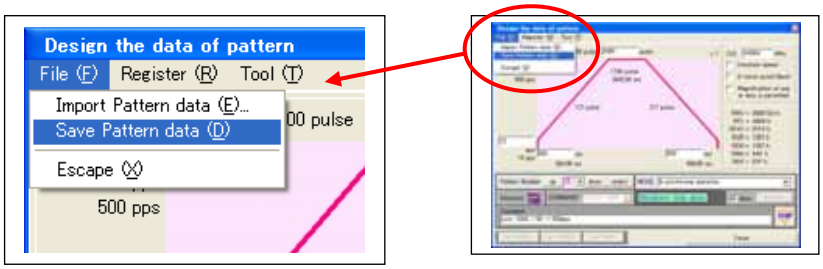
ここまでで作成し、登録したデータを、ファイルとして保存します。保存しないままプログラムを終了すると、登録した全ての動作パターンが失われます。

保存手順は 2 種類あります。どちらか一方を実施してください。

[動作パタンの保存手順 1]

動作パターン設計画面のメニューバーを次のように操作します。

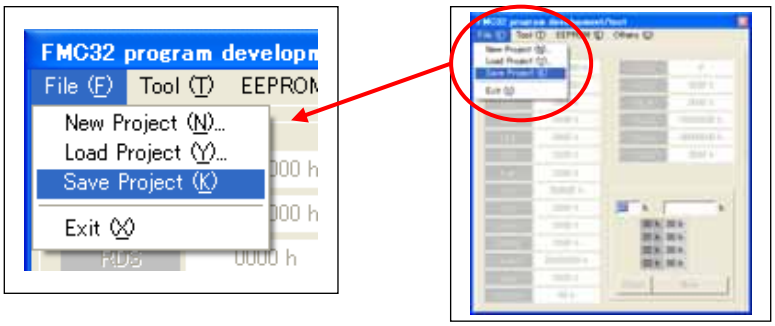
"ファイル" (File) → "パターンデータの保存" (Save Pattern data)をクリックします。



[動作パタンの保存手順 2]

メイン画面のメニューバーを次のように操作します。

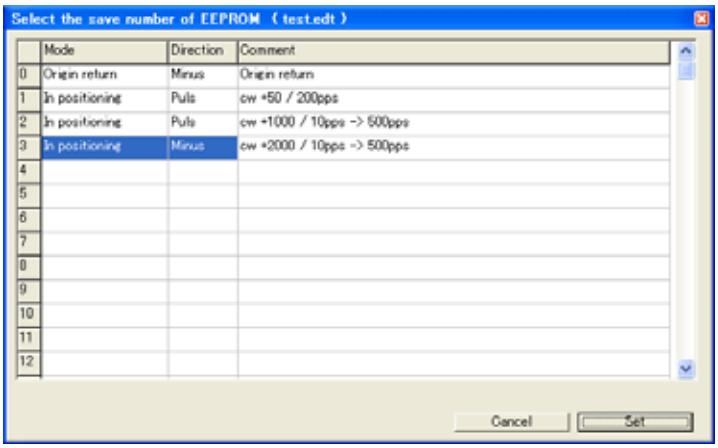
"ファイル" (File) → "プロジェクトの保存" (Save Project)をクリックします。



3-3-7. 登録したデータの確認

どのパターン番号に、何を登録したかを確認できます。
以下の「select」ボタンをクリックしてください。

登録画面が表示され、すでに登録されたパターン番号部分に、その概略が表示されます。



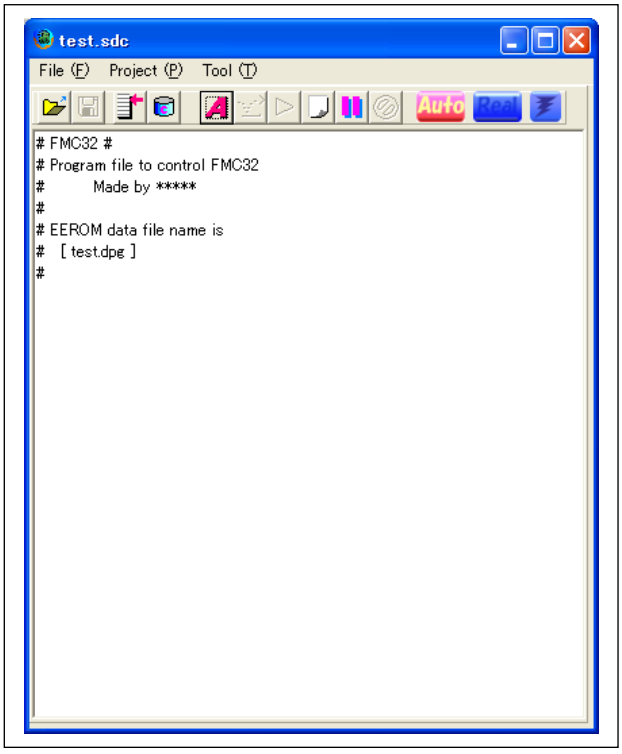
設定したデータの全てが表示されるわけではないので、データの内容はコメントで区別して下さい。そのためにも、動作概要を説明する的確なコメントを入力することを推奨します。

3-4. 動作手順のプログラミング

先に作成した動作パターンを並べて動作手順を設計してゆきます。
設計した動作手順は、思ったとおりの順番で動作するか否かをシミュレーションで確認することができます。

3-4-1. エディタ画面の表示

"ツール" (Tool) メニューの、"プログラム作成" (Design the program)をクリックしてください。
以下のエディタ画面が表示されます。



デフォルトのコメント情報が記述されていますが、これらは削除しないようにお願いします。
また 1 行目は、内容の変更も行わないでください（2 行目以降は変更してかまいません）。

ここで、 "#" で始まる行は全てコメントとして扱われます。
行の途中であっても、1 個以上のスペース文字で区切られた "#" があれば、それ以降の文字列はコメントとして扱われます。

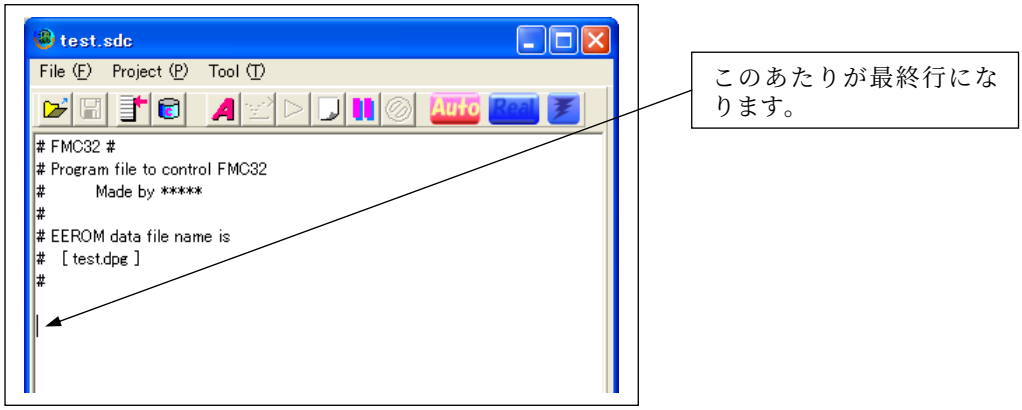
コメント以外の文字列はプログラムとして扱われます。
FMC32 は、単独モードでは、電源が入れられた後、プログラムの最初の行から順番に指定された動作を実行してゆきます。

3-4-2. プログラミング作業の流れ

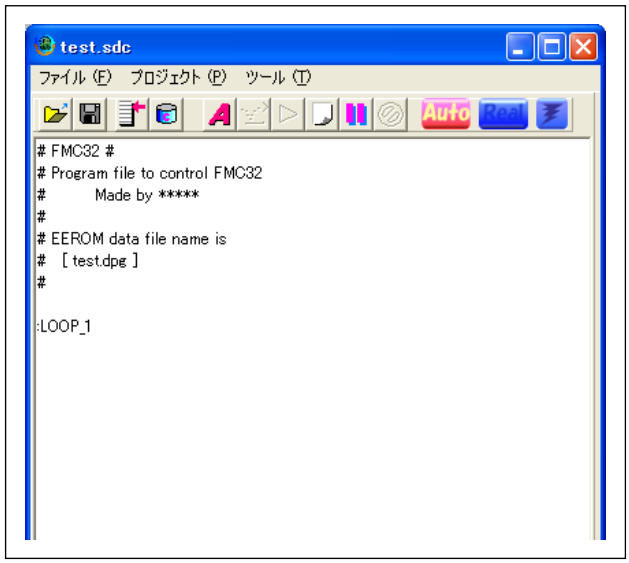
「2-1. 今回作成する動作」で示した動作手順を作成する手順を示します。

[ラベルの設定 1]

今回の動作手順は、定期的に同じ動作を繰り返します。
繰り返し動作のためのジャンプポイントを、 "ラベル" という文字列で指定します。

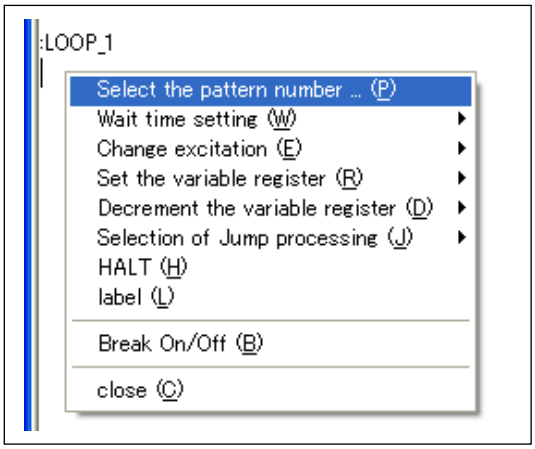


エディタの最終行をクリックし、カーソルを最終行に合わせてください。
ここでキーボードから ":LOOP_1" と入力します。最初のコロン(:)を忘れないでください。
行の先頭にコロンをつけることで、そこに記載された文字列がラベルであることを表します。
ラベル文字は英数字とアンダースコアだけで構成してください。

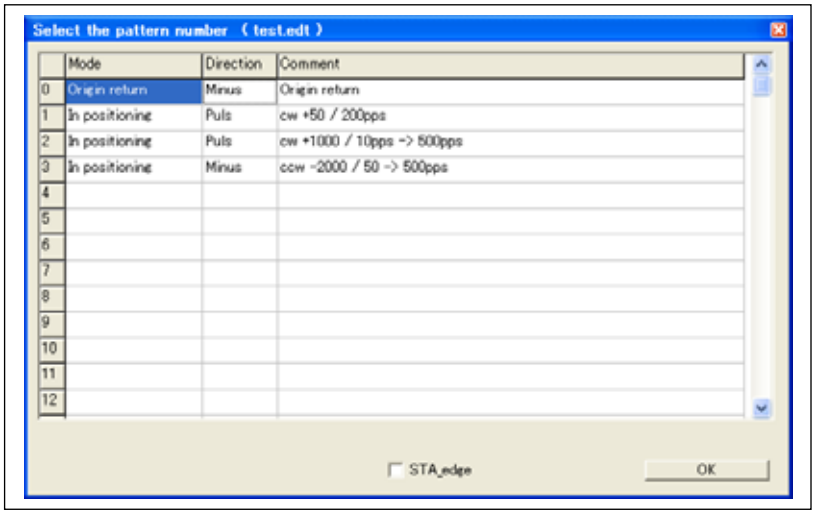


[原点復帰]

ラベルの下の行にカーソルを合わせ、そこで右クリックし、以下のポップアップメニューを表示させます。このポップアップメニューの "パターン番号の選択" (Select the pattern number...) をクリックします。



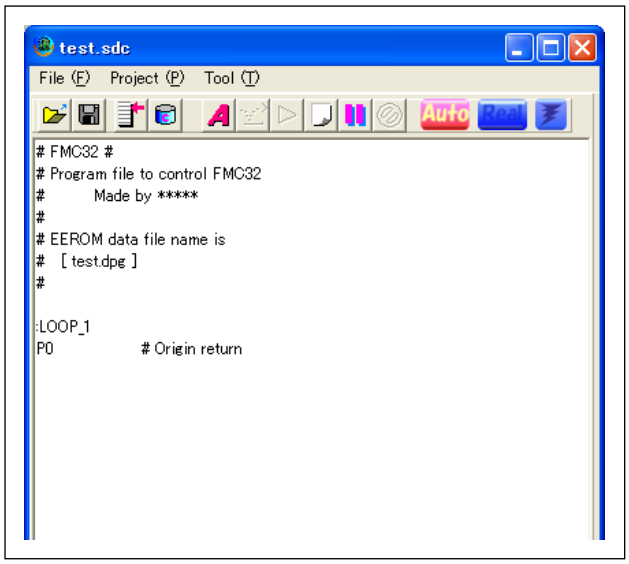
先に作成し、登録した動作パターンの一覧が表示されます。この中からゼロ番の原点復帰 (Origin return)を選択し、 "OK" をクリックしてください。



以下のように選択した動作パターン（パターン番号＝ゼロ）を実行するプログラムが挿入されます。

"P0" が、パターン番号ゼロを実施することを意味しています。

後半の "#" 以降の文字列は、動作パターン作成時に設定したコメントです。

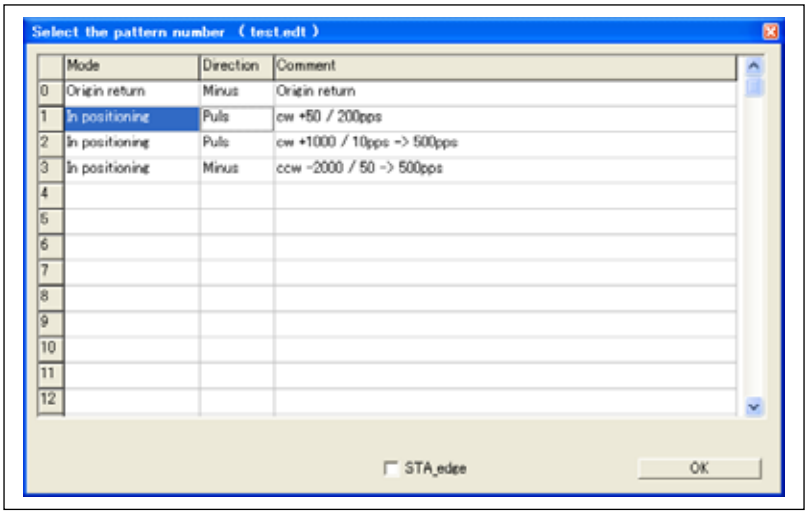


[CW 方向へ 50 パルス出力する動作]

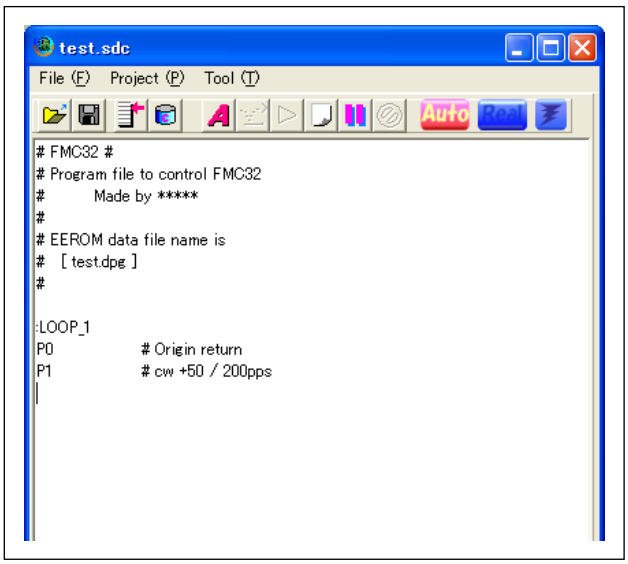
"P0" の下の行にカーソルを合わせ、右クリックします。

ポップアップメニューの "パターン番号の選択" (Select the pattern number...) をクリックします。

動作パターンの一覧が表示されます。この中から１番の位置決め動作(In positionng)を選択し、"OK" をクリックしてください。



次のように "P1" と、そのコメントが挿入されます。

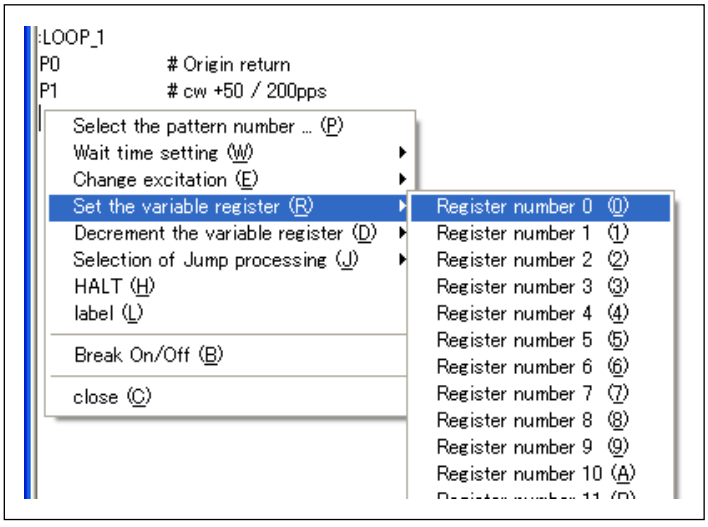


[繰り返し回数の設定]

正転～逆転までのワンセット動作を 3 回繰り返すため、繰り返し回数を設定します。

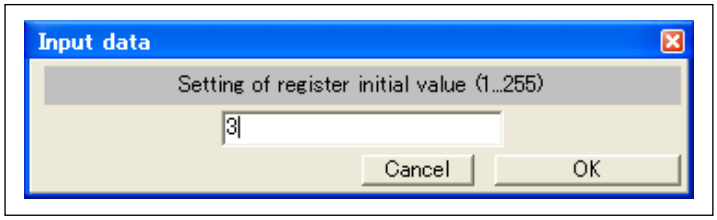
"P1" の下の行にカーソルを合わせ、右クリックします。

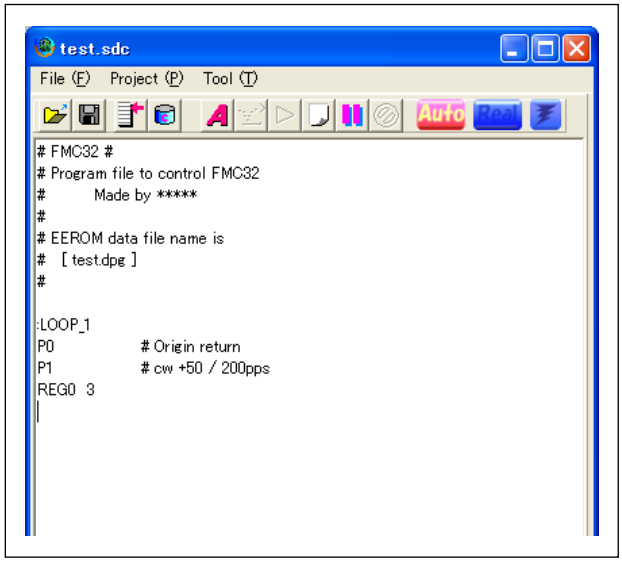
ポップアップメニューの "変数への値設定" (Set the variable register...) をクリックします。さらに次のように "変数番号 0" (Register number 0)をクリックしてください。



以下の、レジスタ初期値の設定(Setting of register initial value)画面が表示されます。ここに "3" を入力し、 "OK" をクリックしてください。

REG0 レジスタへ 3 を代入するプログラムが挿入されます。



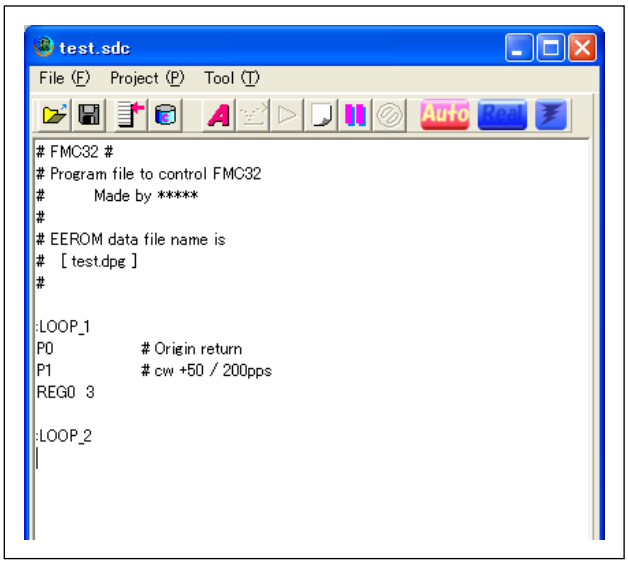


[ラベルの設定 2]

今回の動作は、正転～逆転までのワンセット動作を 3 回繰り返します。この繰り返し動作のためのジャンプポイントを、 "ラベル" として指定します。

"REG0 3" の下の行にカーソルを合わせ、1 回改行します（見た目を良くするためで、必須ではありません。）。

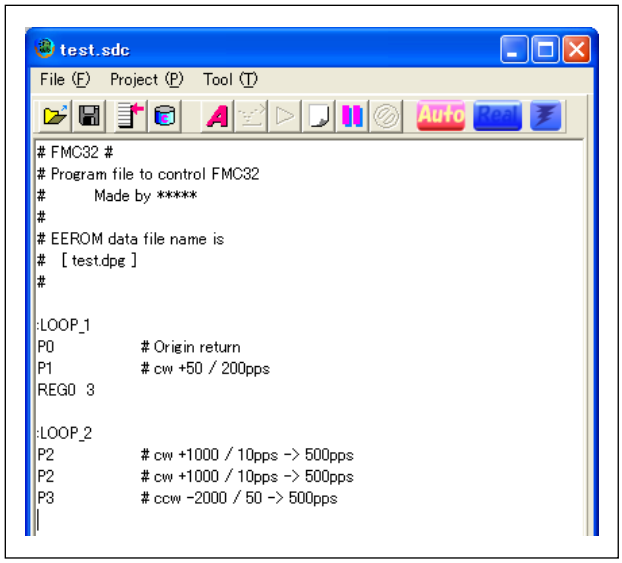
続いて ":LOOP_2" と入力してください。最初のコロン(:)を忘れないでください。



[正転～逆転までのワンセット動作の設定]

今回の動作は、正転～逆転までのワンセット動作を 3 回繰り返します。この繰り返し動作を設定しますが、設定方法は先に説明した "原点復帰" や "CW 方向へ 50 パルス出力する動作" と同じです。

プログラミング後は以下のようになるはずです。

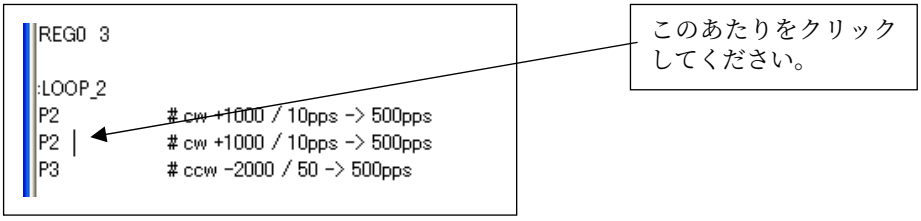


"P2" が 2 行と、 "P3" が 1 行です。

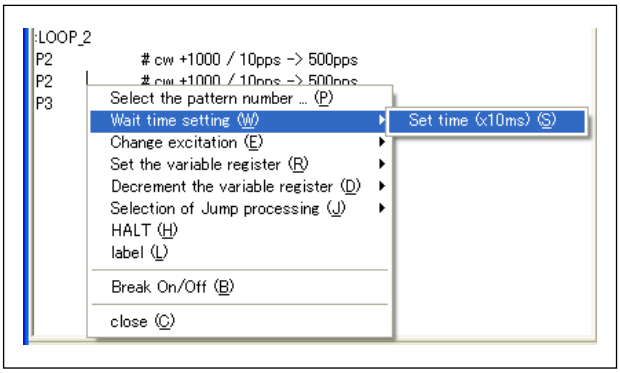
[待ち時間の挿入]

今回の動作は、正転～逆転までのワンセット動作内の各動作終了後、200ms 待ってから次の動作を実行するようにします。

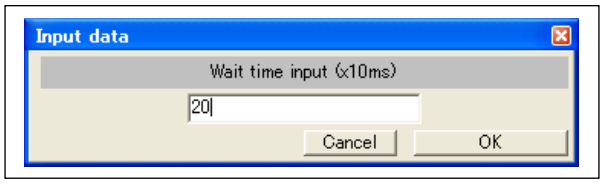
まず 2 つ目の "P2" の行をクリックしてカーソルを合わせてください。



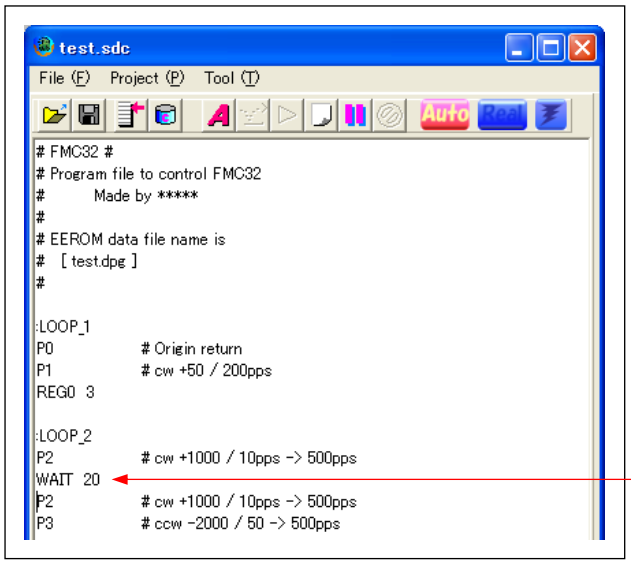
ここで右クリックし、表示されたポップアップメニューの "待ち時間の設定" (Select the pattern number...) をクリックし、さらに "Set time(x10ms)"をクリックしてください。



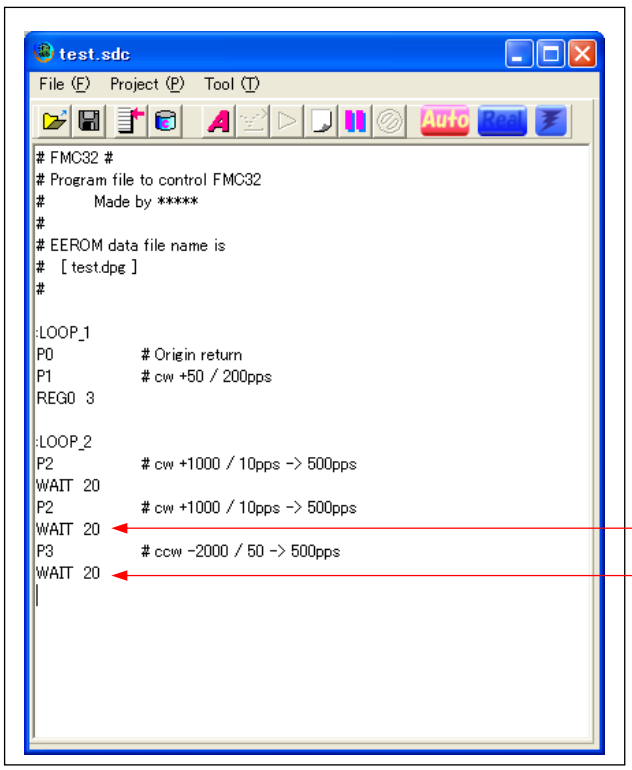
待ち時間の入力(Wait time input)画面が表示されるので、ここに "20" と入力して "OK" をクリックします。ここに入力した値の 10 倍の時間（ミリ秒単位）の待ち時間が発生します。



カーソルで指定した行が１行下へ改行され、 "WAIT 20" というプログラムが挿入されます。



同様な操作で待ち時間を挿入させ、以下のように編集してください。



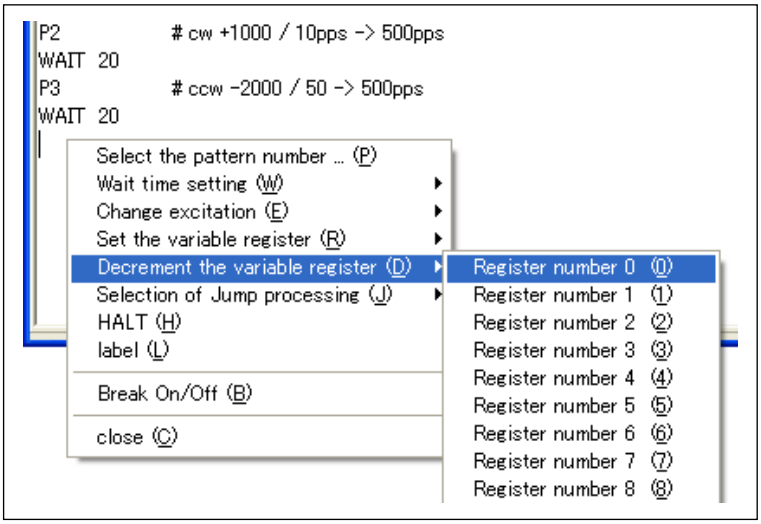
[3 回繰り返す制御]

REG0 の値をデクリメントし、この結果がゼロになったか否かで繰り返し制御を実現します。

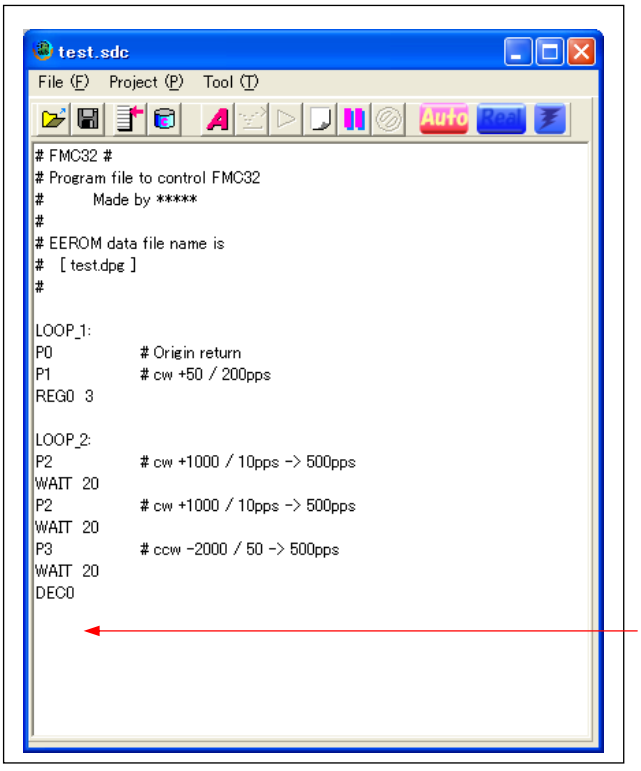
まず REG0 のデクリメントを行います。

3 つ目の "WAIT 20" の下の行をクリックし、カーソルを移動させます。

ここで右クリックし、表示されたポップアップメニューの "変数のデクリメント" (Decrement the variable register) をクリックし、さらに "変数番号 0"(Register number 0)をクリックしてください。


P2 # cw +1000 / 10pps -> 500pps
WAIT 20
P3 # ccw -2000 / 50 -> 500pps
WAIT 20
Select the pattern number ... (P)
Wait time setting (W)
Change excitation (E)
Set the variable register (R)
Decrement the variable register (D)
Selection of Jump processing (J)
HALT (H)
label (L)
Break On/Off (B)
close (C)
Register number 0 (0)
Register number 1 (1)
Register number 2 (2)
Register number 3 (3)
Register number 4 (4)
Register number 5 (5)
Register number 6 (6)
Register number 7 (7)
Register number 8 (8)

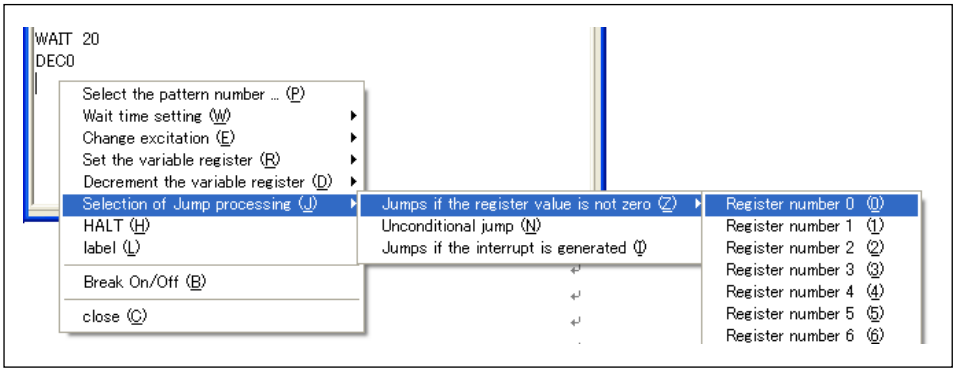
REG0 のデクリメント命令が挿入されます。



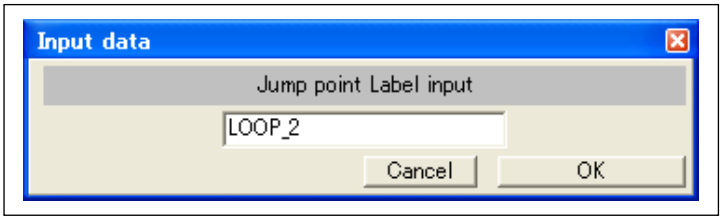
次にジャンプ命令を記述します。

挿入された "DEC0" の下の行にカーソルを移動させ、右クリックし、表示されたポップアップメニューの "ジャンプ処理の選択" (Selection of Jump processing) をクリックし、さらに "変数の値がゼロ以外でジャンプ"(Jumps if the register value is not zero)をクリックし、 "

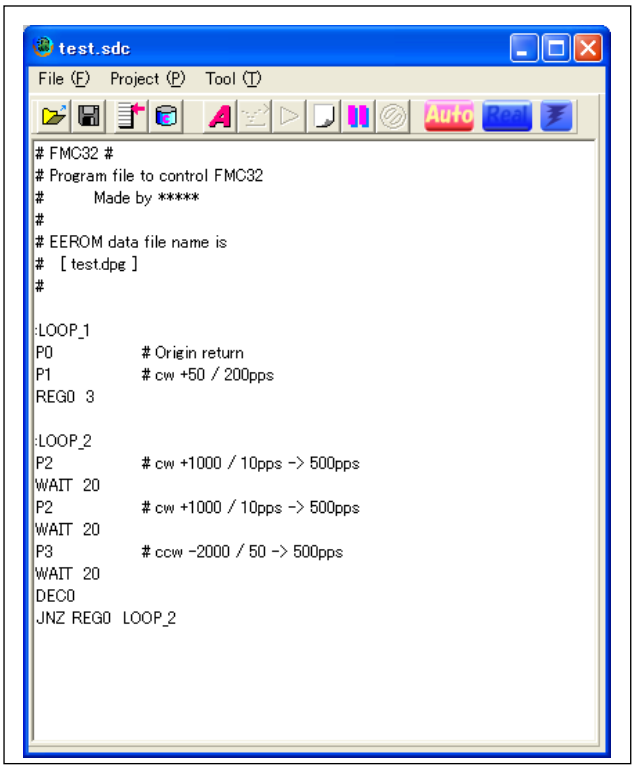
変数番号 0"(Register number 0)をクリックしてください。


WAIT 20
DEC0
Select the pattern number ... (P)
Wait time setting (W)
Change excitation (E)
Set the variable register (R)
Decrement the variable register (D)
Selection of Jump processing (J)
HALT (H)
label (L)
Break On/Off (B)
close (C)
Jumps if the register value is not zero (Z)
Unconditional jump (N)
Jumps if the interrupt is generated (I)
Register number 0 (0)
Register number 1 (1)
Register number 2 (2)
Register number 3 (3)
Register number 4 (4)
Register number 5 (5)
Register number 6 (6)

ジャンプするラベルの入力(Jump point Label input)画面が表示されるので、ここに "LOOP_2" と入力してください。先頭のコロン(:)は不要です。



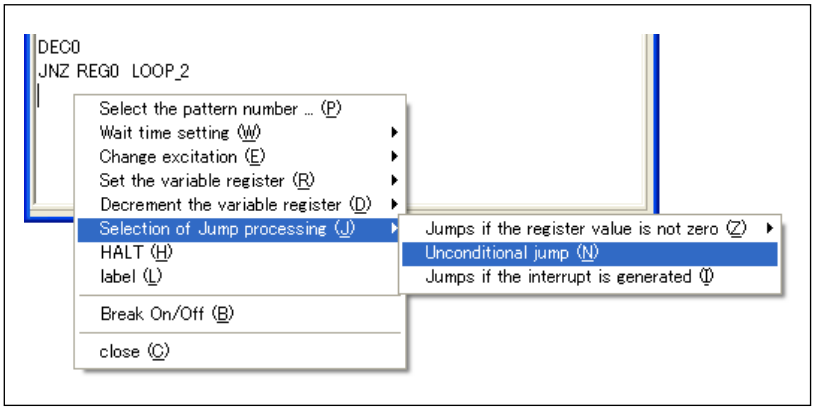
ジャンプ命令が挿入されます。



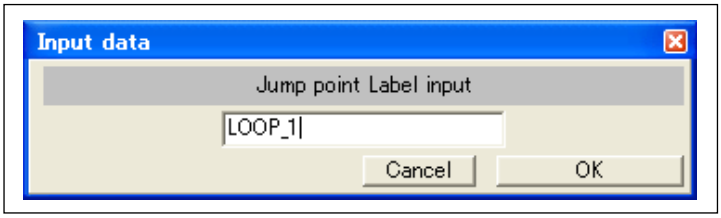
[プログラム先頭へジャンプする処理]

3 回繰り返す動作の完了後、原点復帰動作を再度行うため、プログラムの先頭へジャンプします。

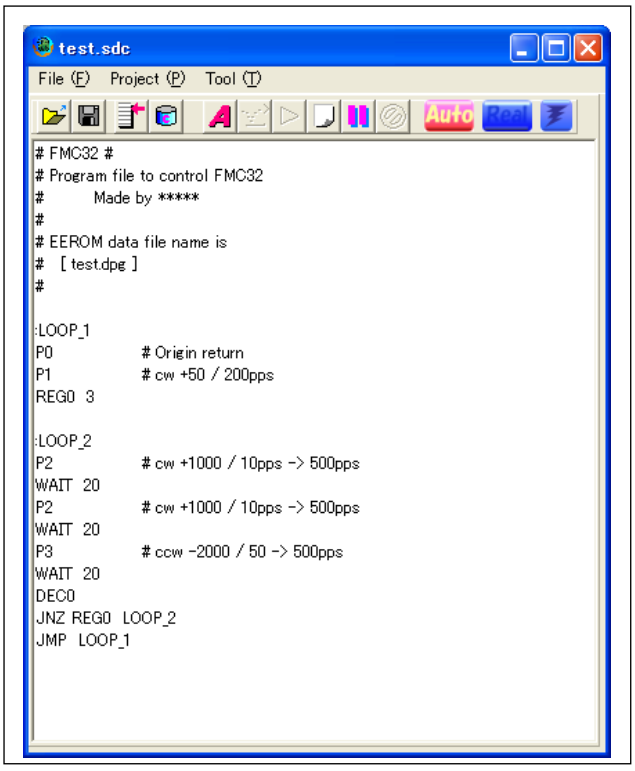
"JNZ REG0 LOOP_2" の下の行にカーソルを移動させ、右クリックし、表示されたポップアップメニューの "ジャンプ処理の選択" (Selection of Jump processing) をクリックし、さらに "無条件ジャンプ"(Unconditional jump)をクリックしてください。


DEC0
JNZ REG0 LOOP_2
Select the pattern number ... (P)
Wait time setting (W)
Change excitation (E)
Set the variable register (R)
Decrement the variable register (D)
Selection of Jump processing (J)
HALT (H)
label (L)
Break On/Off (B)
close (C)
Jumps if the register value is not zero (Z)
Unconditional jump (N)
Jumps if the interrupt is generated (I)

ジャンプするラベルの入力(Jump point Label input)画面が表示されるので、ここに "LOOP_1" と入力してください。先頭のコロン(:)は不要です。



ジャンプ命令が挿入されます。

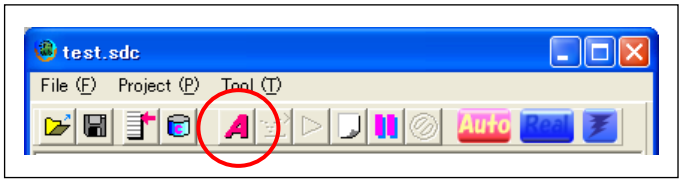


以上で、今回作成する動作のプログラムは完成です。

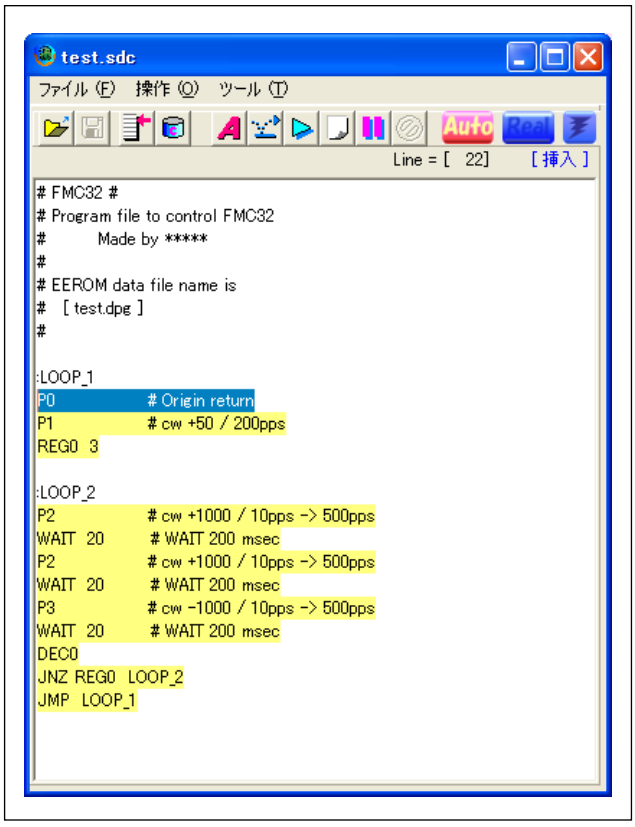
3-4-3. プログラムの動作確認（シミュレーション）

作成したプログラムが、考えたとおりの順番で動作することを確認します。
まずシミュレーションにより、モータを使用しない状態で確認を行います。
シミュレーションによる動作確認は、実際にコマンドが実行されるわけではありません。シミュレーションの目的は、作成したプログラムが思ったとおりの順番で動作することの確認です。

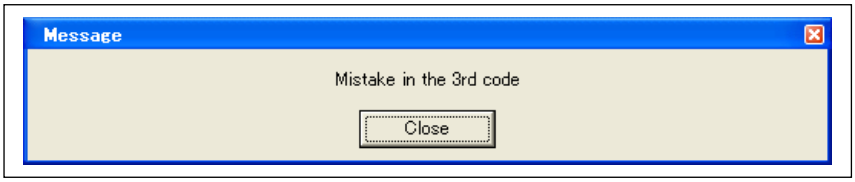
[プログラムのアセンブル]
以下の "アセンブル" ボタンをクリックします。



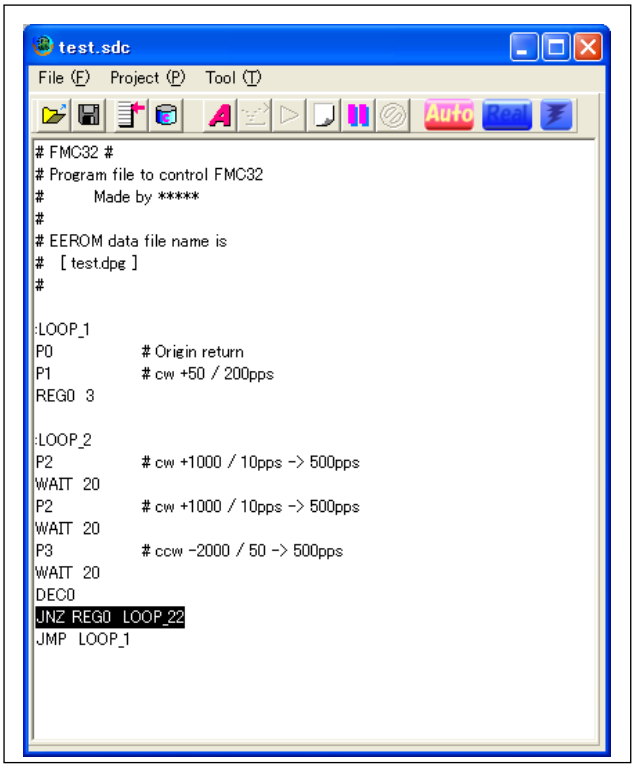
作成したプログラムにエラーが無ければ、以下のようにプログラム部分（コメント、ラベルを除いた文字列）が黄色でハイライトされ、プログラムの先頭行が反転表示となります。



[アセンブルでエラーがあった場合]
仮にプログラムの記述にミスがあると、以下のようなメッセージが表示されます。



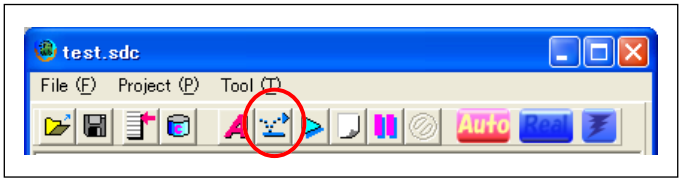
これは命令コードの 3 つ目の文字列が間違っていることを示します。
ここで "Close" をクリックすると、次のように間違いのある行がハイライトされます。



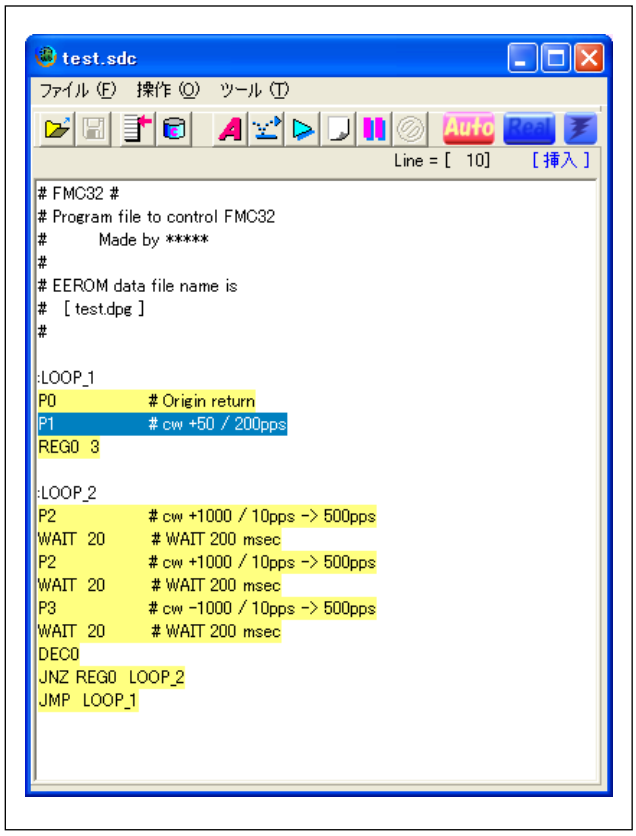
"JNZ REG0 LOOP_22" の記載の 3 番目の文字列は "LOOP_22" です。この記載にミスがあります。

このようにプログラムにミスがある場合、表示されるメッセージと、ハイライト表示される部分を手がかりにして、問題を修正してください。

[ステップ実行による動作確認]
以下の "ステップ実行ボタン" をクリックしてください。



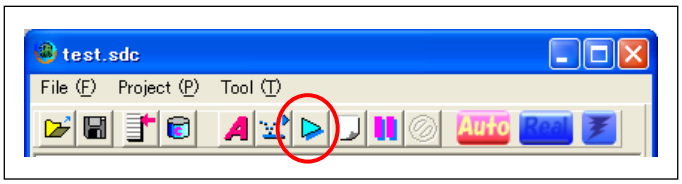
反転表示されているプログラムの 1 行目が実行され、次に実行される予定のプログラム行にハイライト表示が移ります。



以降、"ステップ実行ボタン" をクリックし続けて、プログラムが期待通りの順番になっていることを確認してください。

[連続実行による動作確認]

以下の "実行ボタン" をクリックしてください。

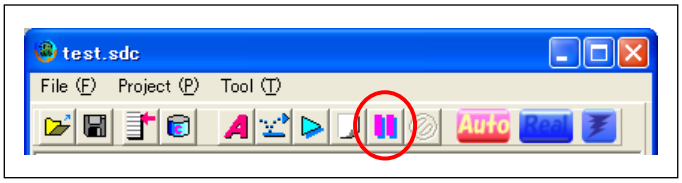


実行されるプログラムが順番にハイライト表示されます。

シミュレーションそのものは、目で追えるくらいの速度で実行されます。

また WAIT コマンドは、この時点では設定した待ち時間は無視され、実際に待つことはありません。

[連続実行の停止]
以下の "停止ボタン" をクリックしてください。



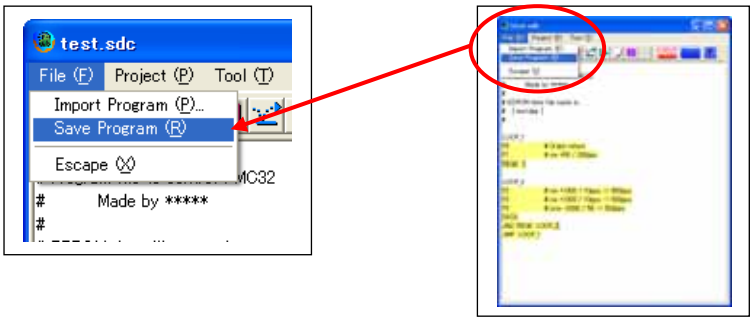
再度 "実行ボタン" をクリックすると、プログラムは停止していた位置から実行されます。

最初から再実行したい場合、再度 "アセンブル" をクリックしてください。

3-4-4. プログラムの保存

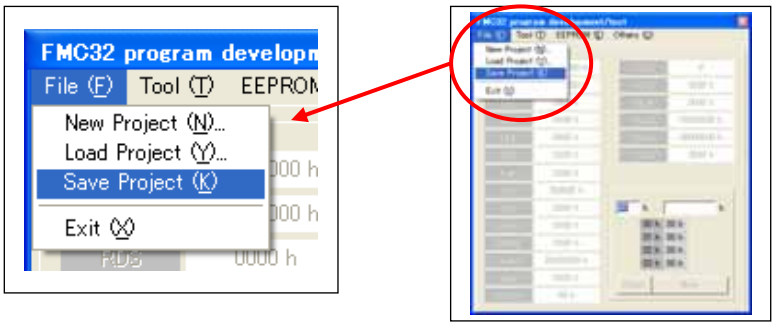
作成したプログラムをファイルとして保存してください。保存しないままプログラムを終了すると、作成したプログラムは失われます。
保存手順は 2 種類あり、どちらか一方を実行してください・

[プログラムの保存手順 1]
エディタ画面のメニューバーを次のように操作します。
"ファイル" (File) → "パターンデータの保存" (Save Pattern data)をクリックします。



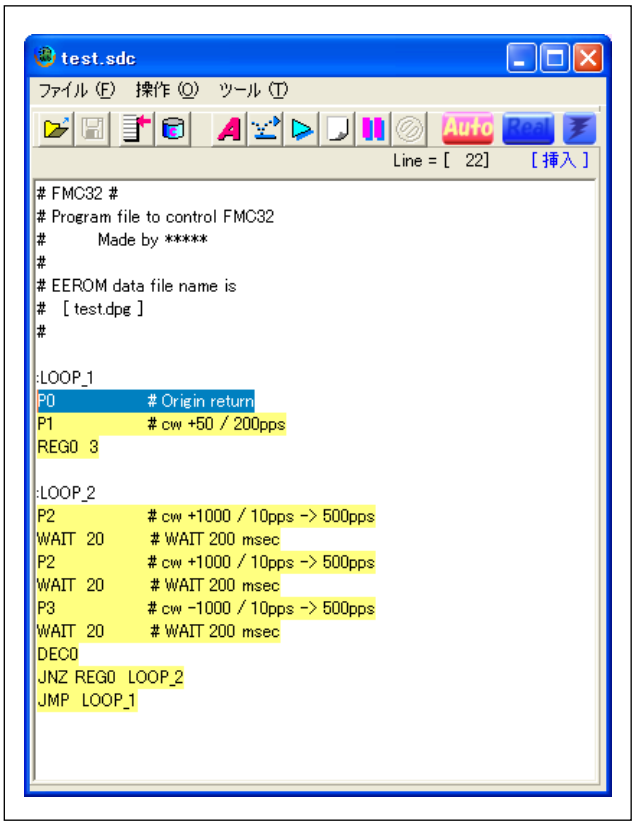
もしくは以下の保存ボタンをクリックしても保存できます。

[プログラムの保存手順 2]
メイン画面のメニューバーを次のように操作します。
"ファイル" (File) → "プロジェクトの保存" (Save Project)をクリックします。



3-4-5. プログラムの動作確認（実動作）

作成したプログラムの動作確認を、実際にモータを接続して行えます。
まずプログラムをアセンブルし、以下の状態としてください。



以下の励磁切替えボタンをクリックしてください。

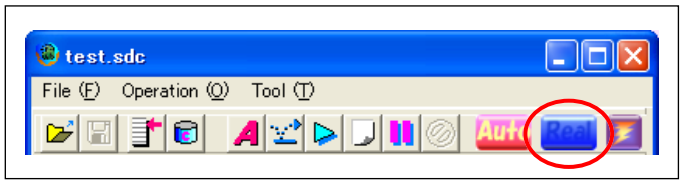
励磁状態が切り替わると、ボタンが次のように変化します。

← 励磁が ON になったことを示します。

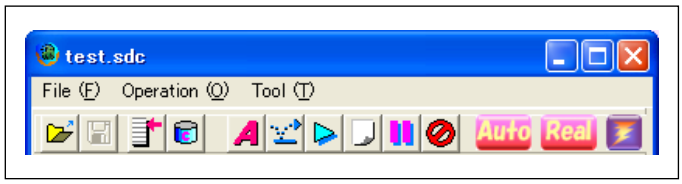
励磁切替えボタンが変化しない場合、励磁状態を変更できないと言うメッセージが表示されます。この場合、FMC32 が接続されていないなどの状況が考えられますので、接続状況や電源などを確認してください。

注：スタートアップガイドでは、励磁切り替えをボタン操作で行っていますが、実際は励磁切り替えを行うコマンドを使用し、プログラム内で自動的に変更できます。
"EX_ON" で励磁 ON、 "EX_OFF" で励磁 OFF に、それぞれ切替えることが可能です。

励磁切替えボタンが変化した後は、以下の REAL ボタン（実動作モードへの切替え）をクリックしてください。



REAL ボタンが以下のように変化すれば、実動作モードになっています。

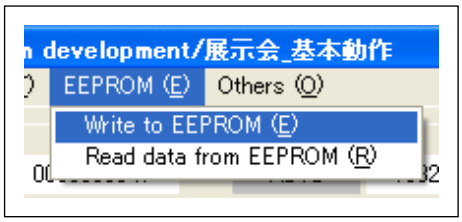


以降の手順は、シミュレーションによる動作確認の時と同じです。

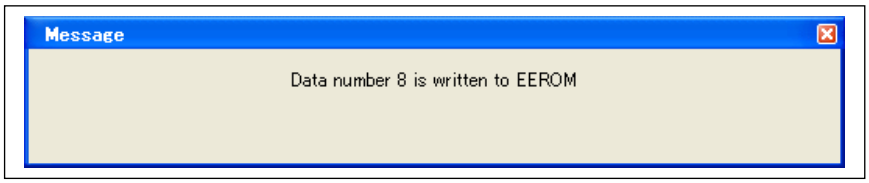
注：PC 制御によって FMC32 を使用している場合、Windows により制御されているため、あるコマンドの終了後、次のコマンドが実行されるまで若干の時間がかかります。そのため、PC 制御での待ち時間(WAIT)は、設定値より若干長くなります。
本ソフトウェアによる FMC32 の制御は、あくまでも動作手順を確認する程度であると考えてください。

3-4-6. EEPROM へのデータの書込み

作成した動作パターンと、動作パターンを利用したプログラムを FMC32 上の EEPROM へ書込みます。
メイン画面の "EEPROM への書込み" (Write to EEPROM)をクリックしてください。



動作パターンを EEPROM へ書込んでいるときは以下のメッセージが表示されます。
また、動作パターンが登録されていない動作番号には書込まれません。またコメントも書込まれません。



動作パターンの書込みが正常に完了すると、続いて実行順序プログラムの書込みが始まります。
全ての書込みが完了すると、以下のメッセージが表示されます。

注意：作成したプログラムは、必ずアセンブルしてから書込んでください。
アセンブルにより、プログラムにエラーがないことを確認してください。
アセンブルしないで書込みを行うと、以下のようなエラーメッセージが表示され、
書込むことができません（動作パターンは書込まれます)。
また未保存の場合も書込みは行えません。からなず保存を実行してください。

メッセージ例
"プログラムをアセンブルしてください"
(Please assemble the program.)
"プログラムが変更されているのでアセンブルしてください"
(The program has been changed. Please assemble it.)
"プログラムを保存してください"
(Please save the program.)
"動作パターンを保存してください"
(Please save the data of pattern.)

3-4-7. スタンドアロンでの実行

まず、USB ケーブルを抜いて、FMC32 と原点スイッチの電源を切ってください。
次に PUSB-3503 からの 4 線式シリアル通信ケーブルを FMC32 から切り離してください。
最後に、先に原点スイッチの電源を入れ、ついで FMC32 の電源を投入してください。

電源投入後、すぐに FMC32 はスタンドアロンモードで、先に書込まれたデータどおりに動作します。
「STA 外部入力」およびに「MON 外部入力」の各ディップスイッチの状況により、動作しない場合があります。
「STA 外部入力」を ON、「MON 外部入力」も ON とすれば動作するはずです。

注意	性能、品質の向上等にともない、お断り無しに掲載事項を変更させていただくことがありますので、あらかじめご了承ください。

NPM 日本パルスモーター株式会社

本社 〒113-0033 東京都文京区本郷 2-16-13
TEL.03-3813-8841 FAX.03-3813-8550
大阪営業所 〒552-0007 大阪府大阪市港区弁天 1-2-1-2314
TEL.06(6576)8330 FAX.06(6576)8335

http://www.pulsemotor.com/

2012 年 4 月 13 日初版作成